

# Estimation of Radio Interferometer Beam Shapes Using Riemannian Optimization

Sarod Yatawatta

Draft version. The final publication is available at [springerlink.com](http://springerlink.com)

**Abstract** The knowledge of receiver beam shapes is essential for accurate radio interferometric imaging. Traditionally, this information is obtained by holographic techniques or by numerical simulation. However, such methods are not feasible for an observation with time varying beams, such as the beams produced by a phased array radio interferometer. We propose the use of the observed data itself for the estimation of the beam shapes. We use the directional gains obtained along multiple sources across the sky for the construction of a time varying beam model. The construction of this model is an ill posed non linear optimization problem. Therefore, we propose to use Riemannian optimization, where we consider the constraints imposed as a manifold. We compare the performance of the proposed approach with traditional unconstrained optimization and give results to show the superiority of the proposed approach.

## 1 Introduction

Most interferometric observations are done using receivers that are more sensitive towards a part of the sky. This narrow field of view is attained using directive antennas (such as a dish) or by beamforming. Due to this reason, images made by such interferometric observations are distorted, with the distortion increasing for celestial objects further away from the direction where the beams are pointed at. Therefore, the knowledge of the beam shape is essential to correct for this distortion while producing accurate and distortion free images. Traditionally beam information is obtained by holographic techniques (Scott and Ryle 1976; Bennet et al 1976; Popping and Braun 2008) or by drift scanning (Pober et al 2011). These methods work well for a stationary and stable beam pattern, such as the beams produced by movable dish based receivers. However, such techniques will not give accurate results for interferometers that have time varying beam shapes. A case in point is the beam shapes produced

---

ASTRON, Dwingeloo, NL  
E-mail: [yatawatta@astron.nl](mailto:yatawatta@astron.nl)

by phased array radio telescopes such as LOFAR<sup>1</sup>. During an observation with a phased array, the beamforming weights change and this results in a variation of the overall beam pattern. In addition, different element layouts between different stations also make the beam shape significantly different. Moreover, secondary effects such as mutual coupling would make the beam shapes different for each receiver.

In this paper, we propose to use the observation itself to extract beam shape information rather than using a priori information such as by holography or drift scanning. Efficient techniques are available to extract directional gains along multiple directions in the sky (Yatawatta et al 2009; Kazemi et al 2011b). We can extract these gains not only along the direction where the beams are pointed at, but also along other directions where there are well known celestial sources to sufficiently sample the beam shape. Once these directional gains are available, the recovery of the beam shape is an ill posed nonlinear optimization problem. Therefore, we have to apply additional constraints to get a satisfactory solution. This naturally leads us to optimization on a Riemannian manifold, as discussed in Gabay (1982); Manton (2004). As a byproduct of this process, we can also obtain intrinsic fluxes of the celestial sources used in calibration, subject to provision of a few known sources for absolute flux calibration.

Manifold optimization has been applied in diverse areas of research and a complete overview is given in Absil et al (2008). In this paper, we present a hybrid optimization method that jointly uses steepest descent (SD) and the Broyden Fletcher Goldfarb Shanno (BFGS) algorithms on a Riemannian manifold. We use the geodesic stepping method (Fiori 2011) based on a Riemannian gradient as our Riemannian steepest descent (RSD) method. However, SD method has the drawback of only having linear convergence rate, especially close to the solution. To accelerate the convergence, we use the Riemannian BFGS (Qi et al 2010) algorithm in conjunction with the RSD method.

Together with calibration along multiple directions (Yatawatta et al 2009; Kazemi et al 2011b), the methods proposed in this paper to estimate the beam shape and intrinsic fluxes can also be considered as one cycle of self-calibration, where we also update the sky model. However, in this paper we focus our attention on the latter part of the self-calibration cycle, i.e., the estimation of beam shapes and the estimation of intrinsic fluxes.

The rest of the paper is organized as follows: In section 2, we give an overview of radio interferometry. Next in section 3, we present the beam shape estimation approach and we elaborate on Riemannian optimization in section 4. In section 5, we give simulation results to verify the proposed approach and give conclusions in section 6.

Notation: Matrices and vectors are denoted by bold upper and lower case letters as  $\mathbf{B}$  and  $\mathbf{v}$ , respectively. The canonical vector with a one at the  $p$ -th location and zeros everywhere else is given by  $\mathbf{e}_p$ . The transpose, Hermitian transpose and conjugation are given by  $(\cdot)^T$ ,  $(\cdot)^H$  and  $(\cdot)^*$ , respectively. The matrix Kronecker product is denoted by  $\otimes$  and the Frobenius norm is given by  $\|\cdot\|$ . The set of complex numbers is denoted by  $\mathbb{C}$ .

---

<sup>1</sup> The Low Frequency Array: <http://www.lofar.org>

## 2 Radio Interferometry

We give a brief overview of radio interferometry and calibration in this section. Consider an interferometer formed by station  $p$  and station  $q$ . The received data after correlation and correction for delay errors between stations can be given as

$$\mathbf{V}_{pq} = \sum_{m=1}^M \mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H + \mathbf{N}_{pq} \quad (1)$$

where  $\mathbf{V}_{pq} (\in \mathbb{C}^{2 \times 2})$  is the visibility matrix (Hamaker et al 1996) and  $\mathbf{N}_{pq} (\in \mathbb{C}^{2 \times 2})$  is the noise. In (1), the observation consists of radiation from  $M$  discrete sources in the sky, whose coherencies are given by  $\tilde{\mathbf{C}}_{pqm} (\in \mathbb{C}^{2 \times 2})$ . For a point source with intensity  $I_m$  and polarized flux  $Q_m, U_m, V_m$ , the coherency for linearly polarized receptors is given by

$$\tilde{\mathbf{C}}_{pqm} = e^{j\phi_{pqm}} \begin{bmatrix} I_m + Q_m & U_m + jV_m \\ U_m - jV_m & I_m - Q_m \end{bmatrix} \quad (2)$$

where  $\phi_{pqm}$  is the Fourier phase component that depends on the direction in the sky as well as the separation of station  $p$  and  $q$ . In calibration, we estimate the Jones matrices  $\mathbf{J}_{pm} (\in \mathbb{C}^{2 \times 2})$  for each station as well as for each direction in the sky (Yatawatta et al 2009; Kazemi et al 2011b). For each source, we have accurate knowledge of the directions (or positions) in the sky and only an apparent knowledge of the fluxes. The solutions obtained for  $\mathbf{J}_{pm}$  contain the information about the beam shape along each direction. However, as noted in Hamaker (2000), there is always an ambiguity in these solutions and therefore, we cannot use the values of  $\mathbf{J}_{pm}$  to directly construct a beam model. Most celestial sources are unpolarized and thus, there is a unitary ambiguity in the solutions and what we obtain is  $\mathbf{J}_{pm} \mathbf{U}_m$  where  $\mathbf{U}_m (\in \mathbb{C}^{2 \times 2})$  is an unknown unitary matrix.

## 3 Beam Shape Estimation

The beam shape of a phased array receiver consists of two parts. Each element used in beamforming has the *element* beam pattern that is sensitive to the full sky. Using beamforming, this beam is narrowed down to cover the field of interest in the sky. Therefore, for the  $p$ -th station, the beam gain along the  $m$ -th direction can be given as  $\gamma_{pm} \mathbf{E}_{pm}$  where the element beam is given by  $\mathbf{E}_{pm} (\in \mathbb{C}^{2 \times 2})$ . What we are interested in is the array gain  $\gamma_{pm} (\in \mathbb{C})$ , which is dependent on the beamforming weights and is changing as the weights change. This is 1 at the direction where the beam is pointed at and as the sky rotates, due to the continuous tracking of a single direction in the sky, the overall values for  $\gamma_{pm}$  change with time.

Note that  $\gamma_{pm}$  is a complex valued parameter: Therefore, in general we estimate the voltage beam shape. Moreover, any atmospheric phase variations are also incorporated into the value of  $\gamma_{pm}$ . Before we proceed, we make the following assumptions:

- We have satisfactory knowledge of the element beam pattern  $\mathbf{E}_{pm}$ , mainly by numerical simulation.

- Although the calibration of (1) was performed using apparent fluxes of the  $M$  sources, we assume approximate knowledge of intrinsic fluxes of at least a few sources (we call them as *seed* sources).
- We assume perfect knowledge of source positions and assume atmospheric phase errors are absorbed into the solutions of  $\mathbf{J}_{pm}$ , in addition to beam shape errors.

We select a set of complex valued basis functions to model the beam shape. Let the number of basis functions be  $D$  and then we can evaluate the beam gain as

$$\gamma_{pm} = \mathbf{e}_p^T \mathbf{B} \mathbf{b}_m \quad (3)$$

where  $\mathbf{B} (\in \mathbb{C}^{N \times D})$  gives the beam model for all  $N$  stations. The values of the basis functions along the  $m$ -th direction is given by  $\mathbf{b}_m (\in \mathbb{C}^{D \times 1})$ . The canonical vector is given as

$$\mathbf{e}_p \triangleq [0, 0, \dots, 0, 1, 0, \dots, 0]^T \quad (4)$$

with all zeros except a 1 at the  $p$ -th location. Ideally, for a source  $m$  with perfect knowledge of its fluxes we get

$$\mathbf{C}_{pqm} \gamma_{pm} \gamma_{qm}^* = \mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H \quad (5)$$

where  $\mathbf{C}_{pqm} (\in \mathbb{C}^{2 \times 2})$  is the true coherency, taking into account the element beam shapes  $\mathbf{E}_{pm}$  and  $\mathbf{E}_{qm}$ . Note that the ambiguities in  $\mathbf{J}_{pm}$  and  $\mathbf{J}_{qm}$  cancel out and does not affect (5).

Using (3) and (5), the cost function that needs to be minimized to estimate  $\mathbf{B}$  can be given as

$$f(\mathbf{B}) = \sum_{p,q,m} \|\mathbf{C}_{pqm} \gamma_{pm} \gamma_{qm}^* - \mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H\|^2 \quad (6)$$

where the summation is taken over all baselines  $p, q$  and all sources  $m$  whose intrinsic fluxes are known. The minimization of (6) to yield an estimate for  $\mathbf{B}$  is highly ill posed mainly due to not having enough sources with known fluxes as well as sufficient intensities to yield a good solution for  $\mathbf{J}_{pm}$  under noisy observations. Therefore, we impose the additional constraint that preserves the total power received by all stations. As shown in appendix A, the total power constraint can be represented as

$$\text{trace}(\mathbf{B}^H \mathbf{B}) = \alpha \quad (7)$$

where  $\alpha$  is a fixed real value. Although we cannot exactly determine  $\alpha$ , a nominal value based on the chosen basis functions and a nominal beam shape is sufficient.

Apart from the cost function (6), we need the gradient of the cost function in our optimization routines. Using techniques of (Hjorungnes and Gesbert 2007), we get the derivative as (proof is given in appendix B)

$$\frac{\partial f}{\partial \mathbf{B}} = \sum_{p,q,m} \beta_{1pqm} \mathbf{\Gamma}_{1pqm} - \beta_{2pqm} \mathbf{\Gamma}_{2pqm} - \beta_{3pqm} \mathbf{\Gamma}_{3pqm} \quad (8)$$

where

$$\begin{aligned}
\beta_{1pqm} &\triangleq \text{trace}(\mathbf{C}_{pqm}^H \mathbf{C}_{pqm}), \\
\beta_{2pqm} &\triangleq \text{trace}(\mathbf{C}_{pqm}^H \mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H), \\
\beta_{3pqm} &\triangleq \text{trace}((\mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H)^H \mathbf{C}_{pqm}), \\
\Gamma_{1pqm} &\triangleq \mathbf{e}_q \mathbf{b}_m^T (\mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_p) (\mathbf{e}_p^T \mathbf{B} \mathbf{b}_m) (\mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_q) \\
&\quad + \mathbf{e}_p \mathbf{b}_m^T (\mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_p) (\mathbf{e}_q^T \mathbf{B} \mathbf{b}_m) (\mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_q), \\
\Gamma_{2pqm} &\triangleq \mathbf{e}_q \mathbf{b}_m^T (\mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_p), \\
\Gamma_{3pqm} &\triangleq \mathbf{e}_p \mathbf{b}_m^T (\mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_q).
\end{aligned} \tag{9}$$

To summarize: we need to estimate  $\mathbf{B}$  by minimizing the cost function (6) subject to the constraint (7). As described in next section, we choose Riemannian optimization to solve this problem.

#### 4 Riemannian Optimization

We give a brief description of the motivation behind using Riemannian optimization as opposed to traditional constrained optimization. As presented in Gabay (1982); Absil et al (2008) and other work, traditional constrained optimization (using Lagrange multipliers) increases the dimensionality of the problem, therefore making it more complicated. On the other hand, the constraints ((7) in our case) can be thought of as restricting  $\mathbf{B}$  onto a Riemannian manifold. Therefore the dimensionality is not increased. However, the traditional gradient based optimization algorithms applicable in Euclidean space cannot be applied in a straight forward manner because the tangent planes change depending on the value of  $\mathbf{B}$ .

We present two optimization algorithms on the manifold (7) that we use in beam estimation. The first one is RSD, as presented in (Fiori 2011) and the second one is RBFGS, as presented in (Qi et al 2010). The steepest descent method has linear convergence but simpler to implement while the RBFGS has super-linear convergence. Apart from the cost function (6), the only requirement is the gradient (8). By using both algorithms in a hybrid fashion, we have faster convergence and are less susceptible to get stuck in a local minimum.

##### 4.1 Riemannian Steepest Descent Algorithm

We choose the method proposed in Fiori (2011) as our RSD algorithm. We briefly give the algorithm that we use for estimating  $\mathbf{B}$ , more detail can be found in Fiori (2011).

1. Calculate  $\frac{\partial f}{\partial \mathbf{B}}$  using (8) and (9).

2. Calculate the Riemannian gradient

$$-\nabla_{\mathbf{B}}f = \frac{\partial f}{\partial \mathbf{B}} - \frac{1}{\alpha} \mathbf{B} \text{real} \left( \text{trace}(\mathbf{B}^H \frac{\partial f}{\partial \mathbf{B}}) \right). \quad (10)$$

3. Find the step size  $h$  in  $[0, 2\pi/\omega]$ ,  $\omega = \|\nabla_{\mathbf{B}}f\|/\sqrt{\alpha}$  that minimizes

$$f(\mathbf{B}(h)) = f(\mathbf{B} \cos(\omega h) + (\nabla_{\mathbf{B}}f) \sin(\omega h)/\omega). \quad (11)$$

4. Update  $\mathbf{B} \leftarrow \mathbf{B} \cos(\omega h) + (\nabla_{\mathbf{B}}f) \sin(\omega h)/\omega$ .

5. If  $\|\nabla_{\mathbf{B}}f\|$  is too small or the maximum number of iterations has reached stop, else go back to step 1.

#### 4.2 Riemannian Broyden Fletcher Goldfarb Shanno Algorithm

In order to present the RBFGS algorithm, we use an alternative representation for the beam model  $\mathbf{B}$  as follows

$$\mathbf{x} \triangleq [\text{vec}(\text{real}(\mathbf{B}))^T \text{vec}(\text{imag}(\mathbf{B}))^T]^T / \sqrt{\alpha} \quad (12)$$

where  $\mathbf{x}$  is a real vector of size  $2ND \times 1$ . Then, the constraint (7) can be rewritten as

$$\mathbf{x}^T \mathbf{x} = 1 \quad (13)$$

which makes  $\mathbf{x}$  restricted to a (real) Stiefel manifold of size  $2ND \times 1$  and dimension  $2ND - 1$ . This also means that  $\mathbf{x}$  is on a  $2ND - 1$  dimensional unit sphere (which is a special case of a Stiefel manifold). We adopt the BFGS algorithm on a unit sphere as presented in Qi et al (2010). In order to fully implement this algorithm, we need to define several operators on the manifold. The projection of any vector  $\boldsymbol{\eta}$  to tangent space at  $\mathbf{x}$  on the manifold is given by

$$\mathbf{P}_{\mathbf{x}}(\boldsymbol{\eta}) \triangleq (\mathbf{I} - \mathbf{x}\mathbf{x}^T)\boldsymbol{\eta}. \quad (14)$$

The cost function (6) can be expressed as  $f(\mathbf{x}) = f(\mathbf{B})$  with some abuse of notation. The gradient is constructed from (8) by projecting it onto the tangent space as

$$\text{grad}(f(\mathbf{x})) = (\mathbf{I} - \mathbf{x}\mathbf{x}^T)[\text{vec}(\text{real}(\frac{\partial f}{\partial \mathbf{B}}))^T \text{vec}(\text{imag}(\frac{\partial f}{\partial \mathbf{B}}))^T]^T / \sqrt{\alpha}. \quad (15)$$

The retraction of vector  $\boldsymbol{\eta}$  in the tangent space at  $\mathbf{x}$  to the manifold is given by

$$\mathbf{R}_{\mathbf{x}}(\boldsymbol{\eta}) \triangleq \frac{(\mathbf{x} + \boldsymbol{\eta})}{\|(\mathbf{x} + \boldsymbol{\eta})\|}. \quad (16)$$

In Qi et al (2010), vector transport is used to transport a tangent vector from a tangent space at one point to the tangent space at another point on the manifold. This operator is given by

$$\mathbf{T}_{\mathbf{x}}(\boldsymbol{\eta}, \boldsymbol{\zeta}) \triangleq \left( \mathbf{I} - \frac{(\mathbf{x} + \boldsymbol{\eta})(\mathbf{x} + \boldsymbol{\eta})^T}{\|(\mathbf{x} + \boldsymbol{\eta})\|^2} \right) \boldsymbol{\zeta} \quad (17)$$

and its inverse (inverse vector transport) is given by

$$\mathbf{T}_{\mathbf{x}}^{-1}(\boldsymbol{\eta}, \boldsymbol{\zeta}) \triangleq \left( \mathbf{I} - \frac{(\mathbf{x} + \boldsymbol{\eta})\mathbf{x}^T}{\mathbf{x}^T(\mathbf{x} + \boldsymbol{\eta})} \right) \boldsymbol{\zeta}. \quad (18)$$

With these definitions at hand, we are ready to implement the RBFSG algorithm.

- Initial conditions: Hessian approximation  $\mathbf{H}_1 = \mathbf{I}$ .
- Iterations  $k = 1$  to *max iterations*
  1. Obtain  $\boldsymbol{\eta}_k$  by solving  $\mathbf{H}_k \boldsymbol{\eta}_k = -\text{grad}f(\mathbf{x}_k)$ .
  2. Perform line search: set  $a = 1$ ;  $c = \text{grad}f(\mathbf{x}_k)^T \boldsymbol{\eta}_k$ 
    - while  $f(\mathbf{R}_{\mathbf{x}_k}(2a\boldsymbol{\eta}_k)) - f(\mathbf{x}_k) < ac$ , update  $a \leftarrow 2a$ .
    - while  $f(\mathbf{R}_{\mathbf{x}_k}(a\boldsymbol{\eta}_k)) - f(\mathbf{x}_k) > 0.5ac$ , update  $a \leftarrow 0.5a$ .
  3. Update  $\mathbf{x}_{k+1} \leftarrow \mathbf{R}_{\mathbf{x}_k}(a\boldsymbol{\eta}_k)$ .
  4.  $\mathbf{s}_k = \mathbf{T}_{\mathbf{x}_k}(a\boldsymbol{\eta}_k, a\boldsymbol{\eta}_k)$ ;  $\mathbf{y}_k = \text{grad}f(\mathbf{x}_{k+1}) - \mathbf{T}_{\mathbf{x}_k}(a\boldsymbol{\eta}_k, \text{grad}f(\mathbf{x}_k))$
  5. Update Hessian approximation as  $\tilde{\mathbf{H}}_k = \mathbf{T}(\mathbf{x}_{k+1}, a\boldsymbol{\eta}_k)\mathbf{H}_k\mathbf{T}^{-1}(\mathbf{x}_{k+1}, a\boldsymbol{\eta}_k)$   
and  $\mathbf{H}_{k+1} = \tilde{\mathbf{H}}_k - \frac{\tilde{\mathbf{H}}_k \mathbf{s}_k \mathbf{s}_k^T \tilde{\mathbf{H}}_k}{\mathbf{s}_k^T \tilde{\mathbf{H}}_k \mathbf{s}_k} + \frac{\mathbf{y}_k \mathbf{y}_k^T}{\mathbf{y}_k^T \mathbf{s}_k}$ .

#### 4.3 Hybrid Optimization

With the RSD and RBFSG algorithms as implemented above, the implementation of the hybrid algorithm is as follows.

1. Start with nominal beam shape  $\mathbf{B}_0$  and  $\alpha = \text{trace}(\mathbf{B}_0^H \mathbf{B}_0)$ .
2. In parallel, run RSD and RBFSG with maximum number of iterations fixed to  $n_1$  (about 10).
3. Compare the final cost from both RSD and RBFSG algorithms. Select the solution with the lowest cost from either RSD or RBFSG as the updated value for  $\mathbf{B}$ .
4. If maximum number of hybrid iterations  $n_2$  (about 200) is reached, stop. Else go back to step 2 with the updated  $\mathbf{B}$  as the initial value.

Note that in this algorithm, we use two limits for the number of iterations, the first one for each RSD and RBFSG iteration limit ( $n_1$ ) and the second one for the hybrid iteration limit ( $n_2$ ). It should also be mentioned that the solution obtained for  $\mathbf{B}$  always has an unknown complex scalar ambiguity. This can be eliminated by normalizing the peak of all the estimated beams to a pure real value.

The initial selection of  $\alpha$  is done by assuming a nominal beam model. Depending on additional information such as the beamformed element layout and the frequency of observation, and also depending on the basis functions chosen, it is possible to determine an accurate value for  $\alpha$ . We also use the nominal beam model as our initial value in optimization.

#### 4.4 Flux Estimation

Once we have the estimate for  $\mathbf{B}$ , it is also possible to estimate the intrinsic fluxes for all sources in our calibration model (1). In order to do this, we make an additional assumption:

- All stations  $p$  see the same intrinsic sky, therefore, for a sky consisting of point sources  $\mathbf{C}_{pqm} = \mathbf{C}_m$  and the common Fourier phase term in (2) and (5) can be precomputed. For an array with parallel dipoles (such as LOFAR), we assume the element beam pattern of each station is identical. Therefore, the dependence of  $p$  and  $q$  on  $\mathbf{C}_{pqm}$  is eliminated.

Under this assumption, for the  $m$ -th source we define the cost to be minimized in order to estimate the flux as

$$g_m(\mathbf{C}_m) = \sum_{p,q} \|\mathbf{C}_m \gamma_{pm} \gamma_{qm}^* - \mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H\|^2. \quad (19)$$

By making  $\frac{\partial g_m(\mathbf{C}_m)}{\partial \mathbf{C}_m} = \mathbf{0}$ , we get the estimate

$$\hat{\mathbf{C}}_m = \frac{\sum_{p,q} \gamma_{pm}^* \gamma_{qm} \mathbf{J}_{pm} \tilde{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H}{\sum_{p,q} |\gamma_{pm}|^2 |\gamma_{qm}|^2}. \quad (20)$$

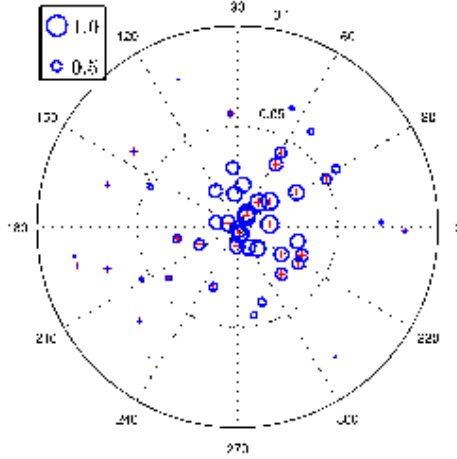
It should be reminded that we can use (20) to estimate fluxes for any point source along the direction of which we have obtained a calibration solution. Of course, for sources that are far away from the center of the beam, the denominator of (20) would get close to zero, making our flux estimate unreliable. This can be overcome by combining observations taken at different epochs. Once we have updated the sky model using (20), we can go back to update our estimate of  $\mathbf{B}$ . Therefore, with an updated sky model, we can use more *seed* sources to better constrain the estimation of the beam shape. In addition, this step also completes one self-calibration loop.

## 5 Simulation Results

We consider an observation with a field of view of 8 degrees (diameter) in the sky. We simulate  $M = 50$  sources, randomly placed in the field of view with no intrinsic polarization  $Q_m = U_m = V_m = 0$  and intensities  $I_m$  varying from 1 to 20 flux units. The positions of the sources are shown in Fig. 1 while the circles sizes indicate the flux ratio between the apparent and true flux values. The number  $M = 50$  was chosen to emulate a typical situation with a LOFAR observation at about 150 MHz with an average beam diameter of about 8 degrees. At much higher frequencies, the beams are narrower and the sources are less bright, therefore, 'clustering' (Kazemi et al 2011a) of sources may be required to get sufficient directions along which to calibrate.

We simulate an interferometer with  $N = 6$  stations. The beam shape of each station is generated to be a Gaussian with random major and minor axes and random offsets from the center of the field of view, as shown in Fig. 2. In addition, we multiply this with a random linear phase screen to make the beam complex. In order to generate the apparent sky model, we attenuate the intrinsic fluxes of the sky model with the mean of the amplitude of the beam shapes shown in Fig. 2, as this is what the fluxes that will be seen in an image made by this interferometer. The mean beam shape is shown in Fig. 3. We also corrupt the apparent fluxes with Gaussian noise, having zero mean and a variance of 0.01.





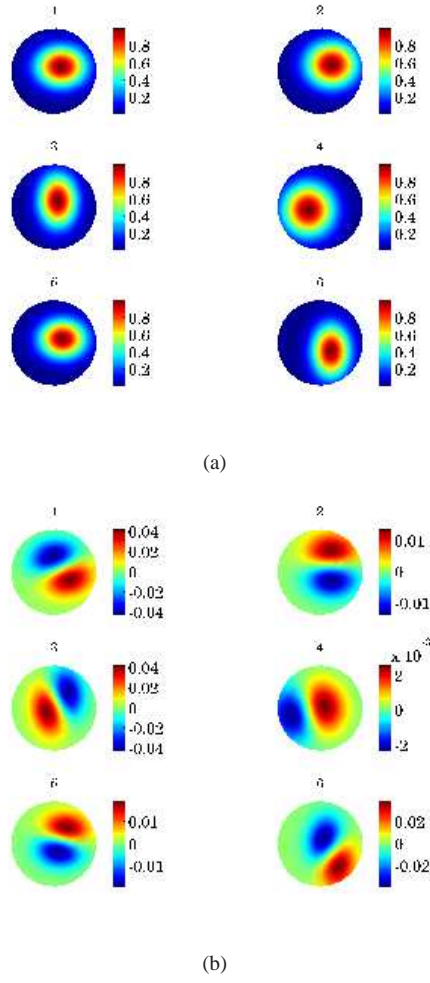
**Fig. 1** Sky model in a field of view of 8 degrees in diameter. The circles correspond to the ratio between the apparent flux used in calibration and the intrinsic flux of each source.

Once we have generated the apparent sky model, we calculate the gain along each direction using the true beam shape and the apparent flux. As an example, we give the gain variation along an azimuthal track for a direction 4 degrees away (in zenith angle) from the field center in Fig. 4. We only show the (1, 1) entry of the matrix  $\mathbf{J}_{pm}$  in Fig. 4. For each direction  $m$ , we calculate the true Jones matrix  $\mathbf{J}_{pm}$  and use a randomly generated unitary matrix  $\mathbf{U}_m$  to get the values used in (6) as  $\mathbf{J}_{pm}\mathbf{U}_m + \mathbf{N}$ , where  $\mathbf{N}$  is a complex Gaussian noise matrix, with elements having zero mean and a variance of 0.01. For a real observation, this step is replaced by calibration along the direction of each source in the sky model.

We consider minimizing (6) by the proposed method as well as by the unconstrained Broyden Fletcher Goldfarb Shanno (BFGS) optimization routine (Nocedal and Wright 1999). For both routines, we need to supply an initial value for  $\mathbf{B}$ . We consider the initial beam for all stations to be a circular Gaussian with major and minor axes diameter of 4 degrees (half the field of view). We selected spherical harmonics with order 4 as the basis functions for  $\mathbf{B}$ . Therefore, there are  $D = 16$  basis functions and the size of  $\mathbf{B}$  is  $6 \times 16$ . The initial value of  $\mathbf{B}$  was used to calculate the value for  $\alpha = \text{trace}(\mathbf{B}^H\mathbf{B})$ . Spherical polar coordinates, centered at the pole of Fig. 1 are used to calculate the basis functions.

The reduction of the cost function with the number of hybrid iterations of the proposed algorithm is shown in Fig. 5. At certain points of the iteration, RBFGS algorithm finds successful solutions and the cost is reduced at a rate which is super-linear.

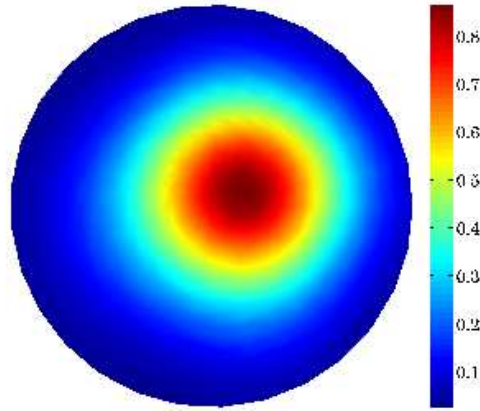
In Fig. 6, we have given the results of the unconstrained optimization with 2500 iterations. The results of the proposed algorithm is shown in Fig. 7, after 250 hybrid



**Fig. 2** Original beams: (a) Real value (b) Imaginary value. Each beam is a Gaussian in amplitude with a randomly chosen major and minor axis and a random offset from the center. The amplitude is multiplied by a random linear phase screen to make the beam complex.

iterations. The inner iterations used is 10 so the total number of iterations for the proposed method is 2500 as well.

Comparison of Figs. 6 and 7 with the original in Fig. 2 clearly shows the superiority of the proposed method. The real values of the beams in Figs. 6 and 7 indicate that the proposed method gives a more focused beam shape as opposed to the unconstrained approach. Moreover, the proposed method recovers the imaginary value of the beams better than the unconstrained approach. Note that the beam number 4 in Fig. 2 has almost zero imaginary value (implying that the phase component is negligible). While the proposed approach also gives a very small value for this beam in Fig. 7, the unconstrained approach gives a significantly higher value, as seen in Fig.



**Fig. 3** Beam shape used for calculating the apparent sky fluxes. We start with a real beam so the imaginary value is zero.

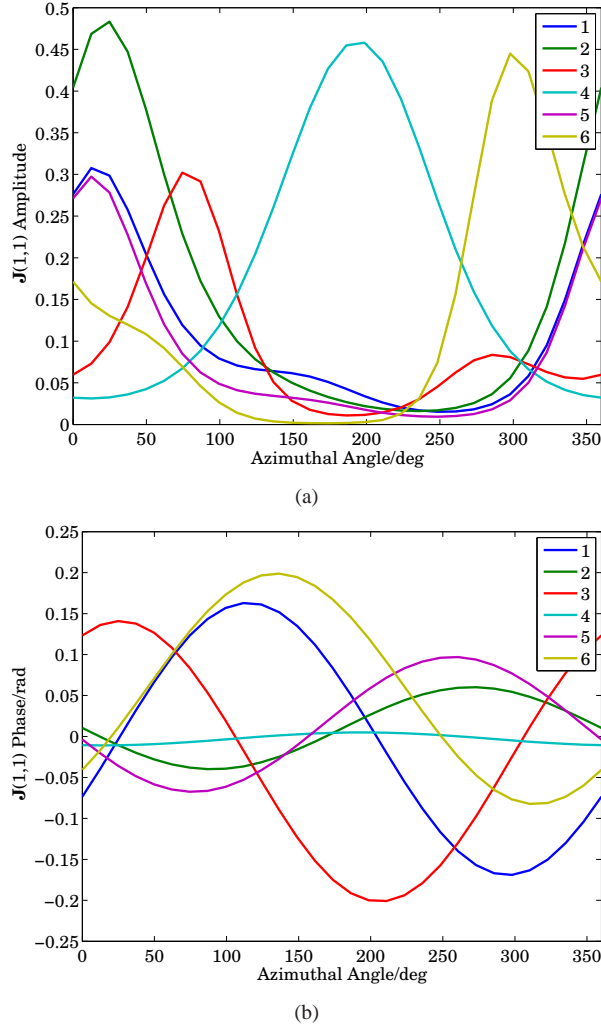
6. In Fig. 8, we also show the error amplitude between estimated beams and the original beams. For a quantitative comparison, we have calculated the total squared error between the original beams and estimated beams for the full field of view, sampled at  $30 \times 30$  grid points. For the unconstrained case, we get a total error of 152 while for the proposed case we only get an error of 130. Comparison of the final cost of  $f(\mathbf{B})$  in (6) at the end of each algorithm shows a different result. With conventional optimization, we get a much lower cost for (6) compared with the proposed method. This is clearly a misleading result due to the ill-posedness of the problem.

In Figs. 9 and 10, we have shown the error in estimating the intrinsic flux using (20). Both figures show the difference of the estimated flux with the true flux by the size of the circles. In Fig. 9, we have used the beam estimate obtained using the unconstrained approach while in Fig. 10 we have used the beam shape obtained using the proposed approach. Both beam estimates give good recovery of the true fluxes within the inner region of the field of view.

The sources at the outlier clearly shows an error in the recovered flux, mainly because their apparent flux is low and are more susceptible to noise. Therefore, in order to improve the flux estimation of outlier sources, we can use diversity in frequency and time. Because the sky is almost invariant, we can combine beam estimates obtained over different time and frequency intervals to improve the flux estimates of outlier sources.

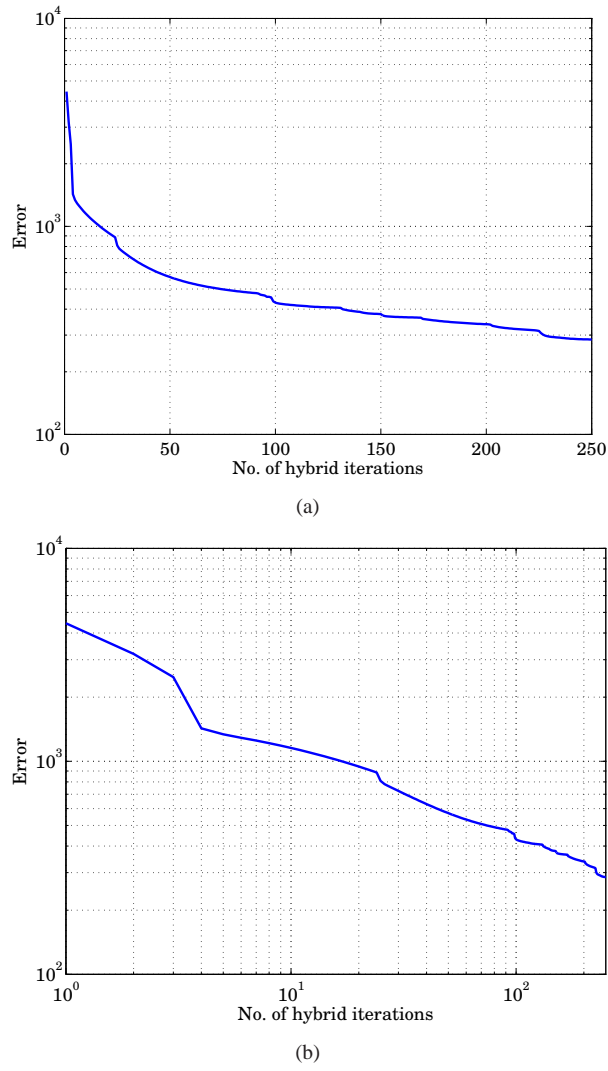
## 6 Conclusions

We have proposed a method of estimating interferometer beam shapes used in a radio interferometric observation by using the directional gains obtained towards known celestial sources. This ill posed problem is solved using optimization on a Rieman-



**Fig. 4** Variation of the beam gain along an azimuthal track: (a) Amplitude (b) Phase. The beam gain is calculated for all 6 stations along a direction 4 degrees away from the center of the field of view.

nian manifold. As compared with conventional optimization, the proposed method give better results. However, the proposed method is computationally more expensive than conventional (unconstrained) optimization. Future work will address the application of this method to real interferometric observations and reducing the computational cost.



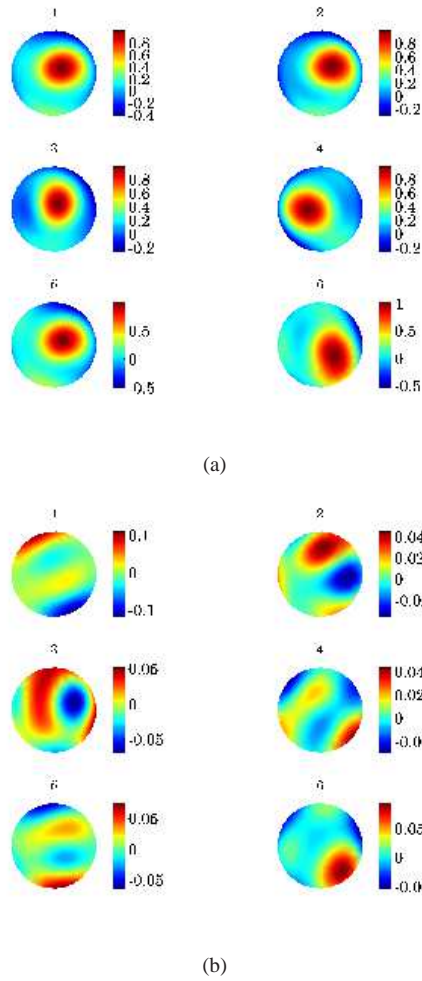
**Fig. 5** Reduction of the cost function with the number of iterations for the proposed algorithm. (a) linear-log scale (b) log-log scale. The sudden jumps of the cost occur when RBFGS finds a successful solution.

## 7 Acknowledgments

We thank the anonymous reviewers for the careful review and helpful comments that enabled us to enhance this paper.

## References

Absil PA, Mahony R, Sepulchre R (2008) Optimization Algorithms on Matrix Man-



**Fig. 6** Estimated beams using unconstrained BFGS optimization: (a) Real value (b) Imaginary value. The total squared error between the original beams and the estimated ones is about 152.

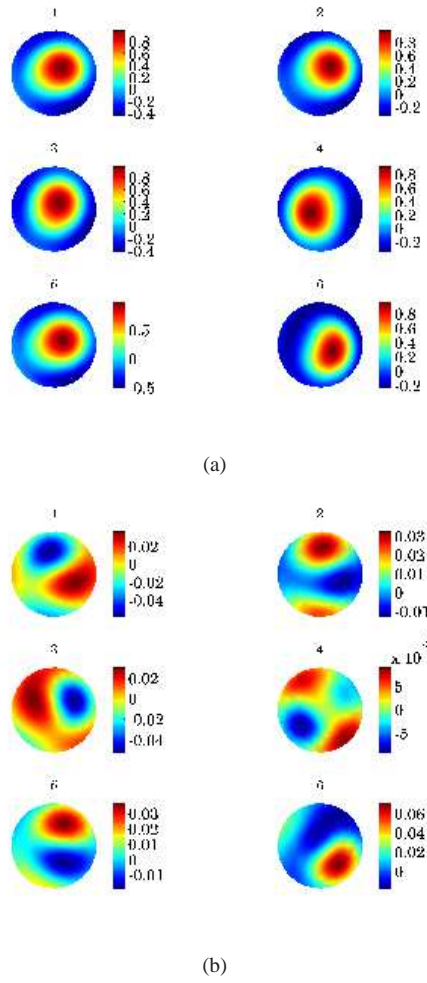
ifolds. Princeton Univ. Press, Princeton NJ

Bennet J, Anderson A, McInnes P, Whitaker J (1976) Microwave holographic metrology of large reflector antennas. *IEEE Trans on Antennas and Propagation* AP-24, no. 3:295–303

Fiori S (2011) Riemannian gradient based learning on the complex matrix hyperspace. *IEEE Trans on Neural Networks* 22, no. 12:2132–2138

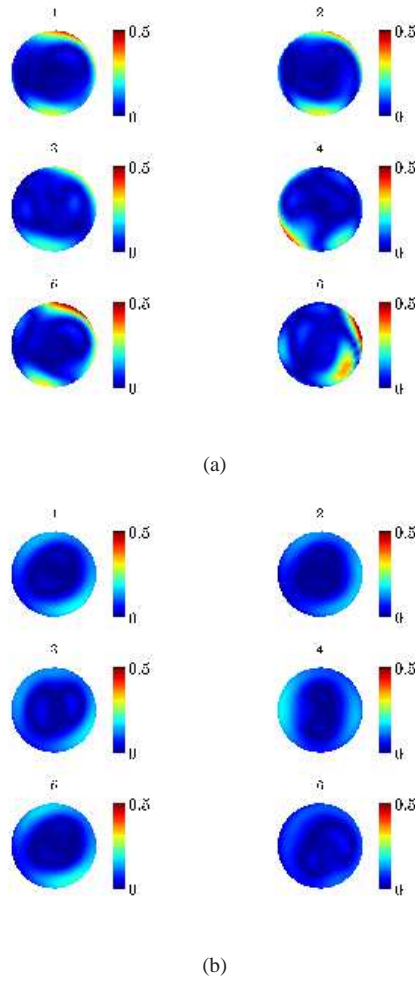
Gabay D (1982) Minimizing a differentiable function over a differential manifold. *Jnl Optim Theory and Appl* 37, no. 2:177–219

Hamaker JP (2000) Understanding Radio Polarimetry IV: The full-coherency analogue of scalar selfcalibration. *Astronomy and Astrophysics Supp* 143(3):515–534



**Fig. 7** Estimated beams using the proposed algorithm. (a) Real value (b) Imaginary value. The total squared error between the original beams and the estimated ones is about 130.

- Hamaker JP, Bregman JD, Sault RJ (1996) Understanding radio polarimetry, paper I. *Astronomy and Astrophysics Supp* 117(137):96–109
- Hjorungnes A, Gesbert D (2007) Complex valued matrix differentiation: Techniques and key results. *IEEE Trans on Sig Proc* 55, no. 6:2740–2746
- Kazemi S, Yatawatta S, Zaroubi S (2011a) Clustered radio interferometric calibration. in *proc IEEE Statistical Signal Processing Workshop (SSP)*, Nice, France
- Kazemi S, Yatawatta S, Zaroubi S, Labropoluos P, de Bruyn G, Koopmans L, Noordam J (2011b) Radio interferometric calibration using the SAGE algorithm. *MNRAS* 414, no. 2:1656–1666
- Manton J (2004) On the various generalisations of optimisation algorithms to manifolds. in *proc Sixteenth International Symposium on Mathematical Theory of Net-*



**Fig. 8** Error amplitude between estimated and original beams. (a) Unconstrained estimate (b) Proposed estimate. There are fewer outliers in the error of the proposed approach.

works and Systems, Belgium

Nocedal J, Wright SJ (1999) Numerical Optimization. New York USA:Springer

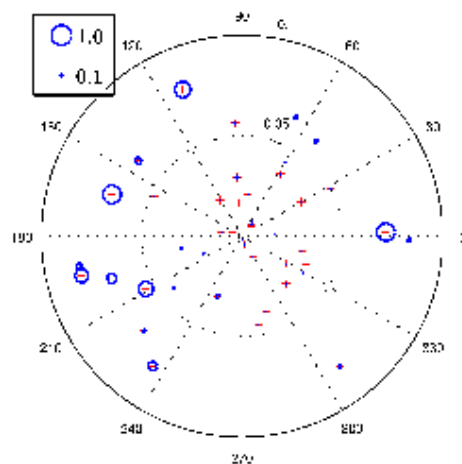
Pober JC, Parsons AR, Jacobs DC, Aguirre JE, Bradley RF, Carilli CL, Gugliucci NE, Moore DF, Parashare CR (2011) A technique for primary beam calibration of drift-scanning, wide-field antenna elements. AJ (submitted)

Popping A, Braun R (2008) The standing wave phenomenon in radio telescopes: Frequency modulation of the WSRT primary beam. A&A 479, no. 3:903–913

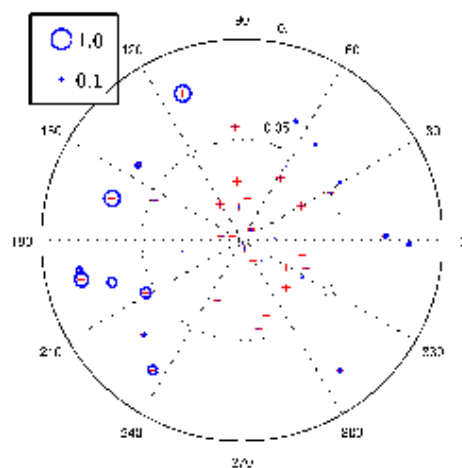
Qi C, Gallivan K, Absil PA (2010) Riemannian BFGS algorithm with applications. Recent Advances in Optimization and its Applications in Engineering pp 183–192

Scott P, Ryle M (1976) A rapid method for measuring the figure of a radio telescope reflector. MNRAS 178:539–545





**Fig. 9** The error (difference between estimated flux and true flux) in estimated fluxes using the beam shape obtained by unconstrained optimization. The size of the circles represent the magnitude of the absolute error.



**Fig. 10** The error (difference between estimated flux and true flux) in estimated fluxes using the beam shape obtained by the proposed approach. The size of the circles represent the magnitude of the absolute error.

Yatawatta S, Zaroubi S, de Bruyn G, Koopmans L, Noordam J (2009) Radio interferometric calibration using the SAGE algorithm. in proc 13th IEEE DSP workshop pp 150–155

### A Proof of (7):

Let the total power received by all stations be  $\Gamma$ . We can express this as

$$\Gamma = \sum_{p,m} |\gamma_{pm}|^2 = \sum_{p,m} |\mathbf{e}_p^T \mathbf{B} \mathbf{b}_m|^2 \quad (21)$$

where  $p$  is summation over all stations  $(1, \dots, N)$  and  $m$  is summation over an infinite number of directions in the sky that covers the full field of view. Note that the summation over  $m$  is not restricted to the directions where we have known sources. We have

$$\Gamma = \text{trace} \left( \mathbf{B} \sum_m (\mathbf{b}_m \mathbf{b}_m^H \mathbf{B}^H \sum_p \mathbf{e}_p \mathbf{e}_p^T) \right) \quad (22)$$

and using the fact that

$$\sum_p \mathbf{e}_p \mathbf{e}_p^T = \mathbf{I} \quad (23)$$

we get

$$\Gamma = \text{trace} \left( \sum_m (\mathbf{b}_m \mathbf{b}_m^H \mathbf{B}^H \mathbf{B}) \right). \quad (24)$$

Let

$$\boldsymbol{\Upsilon} \boldsymbol{\Upsilon}^H = \sum_m (\mathbf{b}_m \mathbf{b}_m^H). \quad (25)$$

Then,

$$\Gamma = \text{trace} \left( (\mathbf{B} \boldsymbol{\Upsilon})^H (\mathbf{B} \boldsymbol{\Upsilon}) \right) = \|\mathbf{B} \boldsymbol{\Upsilon}\|^2 \leq \|\mathbf{B}\|^2 \|\boldsymbol{\Upsilon}\|^2 \quad (26)$$

Taking into account that  $\|\boldsymbol{\Upsilon}\|^2$  is fixed for a given basis, we can keep  $\Gamma$  below a certain level by keeping

$$\|\mathbf{B}\|^2 = \text{trace} (\mathbf{B}^H \mathbf{B}) = \alpha \quad (27)$$

where  $\alpha$  is a fixed real value. One additional point to be raised here is that by selecting an orthonormal basis, we get  $\boldsymbol{\Upsilon} \approx \mathbf{I}$ , therefore an orthonormal basis is always preferred (although in practice hard to realize).

### B Proof of (8):

We can rewrite (6) as

$$f(\mathbf{B}) = \sum_{p,q,m} \text{trace} \left( (\mathbf{X}_{pqm} \otimes \mathbf{Z}_{pqm} - \mathbf{Y}_{pqm} \otimes \mathbf{1})^H (\mathbf{X}_{pqm} \otimes \mathbf{Z}_{pqm} - \mathbf{Y}_{pqm} \otimes \mathbf{1}) \right) \quad (28)$$

where  $\mathbf{X}_{pqm} \triangleq \mathbf{C}_{pqm}$ ,  $\mathbf{Z}_{pqm} \triangleq \mathbf{e}_p^T \mathbf{B} \mathbf{b}_m \mathbf{b}_m^H \mathbf{B}^H \mathbf{e}_q$ ,  $\mathbf{Y}_{pqm} \triangleq \mathbf{J}_{pm} \bar{\mathbf{C}}_{pqm} \mathbf{J}_{qm}^H$  and  $\mathbf{1} \triangleq \mathbf{1}$ . This can be simplified as

$$\begin{aligned} f(\mathbf{B}) = & \sum_{p,q,m} \text{trace}(\mathbf{X}_{pqm}^H \mathbf{X}_{pqm}) \text{trace}(\mathbf{Z}_{pqm}^H \mathbf{Z}_{pqm}) \\ & - \text{trace}(\mathbf{X}_{pqm}^H \mathbf{Y}_{pqm}) \text{trace}(\mathbf{Z}_{pqm}^H) \\ & - \text{trace}(\mathbf{Y}_{pqm}^H \mathbf{X}_{pqm}) \text{trace}(\mathbf{Z}_{pqm}) \\ & + \text{trace}(\mathbf{Y}_{pqm}^H \mathbf{Y}_{pqm}). \end{aligned} \quad (29)$$

Using Hjorungnes and Gesbert (2007), we can take the derivative of each term with a trace of  $\mathbf{Z}_{pqm}$  to yield (8).